

REMARKS/ARGUMENTS

1. Claim Amendments

Claims 23-44 are pending in the application. Favorable reconsideration of the application is respectfully requested in view of the foregoing amendments and the following remarks.

2. Claim Rejections – 35 U.S.C. § 112

The Examiner objected to Claim 39 under 35 U.S.C. § 112 as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. Claim 39 has been amended to overcome the rejection.

3. Claim Rejections – 35 U.S.C. § 103(a)

Claims 23-26, 29-34, 37-39, 41-44 stand rejected under 35 U.S.C. 103(a) as being unpatentable over Kuhn. US Patent No. 6,961,567 (Kuhn) and further in view of Allor, US Patent Application No. 2003/0226102 (Allor). Applicant first notes that the Final Office Action sets forth a new ground of rejection from that presented in the prior Office Action, and hence, the Final Office Action should be Non-Final so as to permit the Applicant a fair opportunity to address the new rejections.

The currently pending independent apparatus claim is claim 23:

23. A mobile terminal for use in a wireless telecommunications system, comprising:

a mobile terminal;

a mobile terminal platform domain having a software services component for providing functionality, said software services component in the form of software instructions adapted to be loaded and stored in a computer readable medium and executed by a processor of the mobile terminal.

the mobile terminal platform domain further having an interface component having at least one interface for providing access to the functionality of the software services component for enabling an application domain software to be installed, loaded and run in said mobile terminal platform via said at least one interface, said interface component in the form of software instructions adapted to be loaded and stored in a computer readable medium and executed by a processor; and

plug-in software in the form of software instructions adapted to be loaded and stored in a computer readable medium and executed by the processor of the mobile terminal for use by the application domain software for modifying the functionality of the software services component of the mobile terminal platform domain via the at least one interface.

The currently pending independent method claim is claim 32:

32. A method for use in a mobile terminal, comprising:

providing a mobile terminal platform domain having a software services component for providing functionality, said software services component in the form of software instructions adapted to be loaded and stored in a computer readable medium and executed by a processor in the mobile terminal;

providing an interface component in said mobile terminal platform domain having at least one interface for providing access to the functionality of the software services component for enabling an application domain software to be installed, loaded and run in said mobile terminal platform via said at least one interface, said interface component in the form of software instructions adapted to be loaded and stored in a computer readable medium and executed by the processor of the mobile terminal;

providing plug-in software in the form of software instructions adapted to be loaded and stored in a computer readable medium and executed by the processor of the mobile terminal and together with the application software for modifying the functionality of the software services component of the mobile terminal platform domain via the at least one interface; and

modifying the functionality of the software services component via said plug-in software.

According to the Examiner, with respect to claims 23 and 32:

Kuhn discloses a mobile terminal for use in a wireless telecommunications system comprising:

a mobile terminal; a mobile terminal platform domain (e.g. Fig. 1)

having a software services component for providing functionality, said software services component in the form of software instructions adapted to be loaded and stored in a computer readable medium (col. 4, lines 2-6)

and executed by a processor of the mobile terminal, the mobile terminal platform domain further having an interface component having at least one interface for providing access to the functionality of the software services component for enabling an application domain software to be loaded and run in said mobile terminal platform via said at least one interface (e.g. icon, launch ... interaction, perform tasks, number of screens - col. 5, lines 24-45),

said interface component in the form of software instructions adapted to be loaded and stored in a computer readable medium (col. 5, line 65 to col. 6, line 6)

and executed by a processor; and

plug-in software in the form of software instructions adapted to be loaded and stored in a computer readable medium and executed by the processor of the mobile terminal for use by the application domain software for modifying the functionality of the software services component (e.g. plug-in 204, Fig. 2; deactivation - col. 6, lines 31-47, lines 57-65; switch between carriers; changing computer's compatibility for different carrier... providers - col. 7, lines 29- 61) of the mobile terminal platform domain via the at least one interface.

The Examiner further states that Kuhn does not explicitly disclose an interface component for enabling application software to be installed:

Kuhn's HTTP application (see Fig. 3, 5A-B and related text) is depicted with extensive use of plug-in being invoked for setting features, downloading (see registration file 208, Table 1, col. 8, col. 9, lines 1-25)

and detecting error conditions (see col. 6, lines 57-65) with regard to a service provider type of activation. This Web-based remote activation

is further enhanced with the analogous use of plug-in by Allor, according to which, a extensible tag functionality from a plug-in module can be used to support installation of components via callback information in a browser interface (installation, para. 0048, pg. 5).

According to the Examiner, it would have been obvious for one skill in the art at the time the invention was made to implement the browser environment plug-in by Kuhn so that the plug-in can support validating and warning of errors in support of installation of components for a particular provider domain application, based on Allor's teachings in using plug-in tag indicator and callback information. Applicant does not agree that there is a motivation for combining Kuhn with Allor based on the Examiner's statement as summarized above. This is all the more so as elements of Kuhn cannot be equated to those of the present invention, as has been done in the Final Office Action and as described below.

According to the Examiner, embedded content or browser functionality in Kuhn's provider-specific Web services can be installed and would be deployed using the ready-to-use or provided plug-in for such browsers, "alleviating thereby resources that would require additional developing of program native code, such that extensibility of browser capabilities and content display had been known to be supported by ready-to-use plug-ins" (The Examiner cites Allor, pg. 1, paragraphs 0004-0005).

Applicant respectfully disagrees with Examiner's characterization of Kuhn. Kuhn actually teaches a system and method for managing wireless network activation with a carrier and for handling registration with a service provider. Kuhn does not disclose a mobile terminal platform domain within a mobile terminal, the platform having an interface component having at least one interface for providing access to the functionality of a software services component for enabling an application domain software to be loaded and run in said mobile terminal platform via said at least one interface. Elements of Kuhn which are equated by the Examiner to certain elements of the present invention do not share the same structure or functionality. For example, the Examiner cites col. 5, lines 24-45:

In one embodiment, the driver device 202 includes an icon on the display screen 130 for a user to select. Selection of the icon launches

GARF 200 procedures, including activating a wireless network with a carrier and handling user registration with a service provider. BellSouth, used with Palm VII, is an example of a carrier. Palm.Net, used with Palm VII, is an example of a service provider. Palm.Net currently resells BellSouth services for use with Palm VII. In one embodiment, the driver device 202 is an executable application that a user will launch upon their first interaction with a wireless capable computer 100. The driver device 202 will drive the configuration of the computer 100 to obtain wireless connectivity, as well as perform the tasks required to obtain an account and or identity with a particular carrier and or service provider. In one example, the driver device 202 may be launched by applications such as Clipper (used in the Palm operating system) when wireless connectivity has not yet been established for the computer 100. A feature of the GARF 200 is the ability for a user to step through a number of screens in order to provide personal data to a wireless service provider, such as Palm.Net.

GARF refers to the architecture for the generic activation and registration framework (GARF) as seen in Figure 3 of Kuhn:

The GARF is a system for providing network activation and registration in a network that may have multiple carriers and or service providers. The GARF 200 provides a driver device 202 that generally controls network activation and registration procedures for the computer 100. Networking applications that may be tailored for a specific carrier or service provider are not included in the driver device. Rather, a plug-in device 204 (or multiple plug-in devices) are installed in the computer 100. A plug-in device 204 includes hardware and or software specifically tailored for a particular carrier and service provider. Through a given plug-in device 204, the computer 100 can communicate with a carrier and service provider that is intended for that plug-in device 204. For example, a driver device 202 may communicate with a provider's web server 206 with assistance from the plug-in device 204. Additionally, the provider's web server 206 can send a registration file 208 to the driver device 202, which can then launch the registration file 208 in the computer 100. The registration file 208 is an executable file configured to allow the computer 100 to be compatible with a particular carrier and or service provider. The GARF 200 allows manufacturers to build a computer 100 for operation in wireless networks of any verbal language. For example, a generic driver device 202 may be built that is nonspecific to any particular verbal language. The plug-in device 204, on the other hand, may be then be configured to be language specific. If France, for instance, has three different service providers that operate the same way in the French language, the GARF 200 allows one computer 100 to communicate with all three service providers. In another embodiment, the plug-in device 204

is generically built such that the computer 100 is generically operable in any wireless network of any verbal language.

As can be seen, the functionality of the elements in the Kuhn invention cannot be equated to those of the present invention, because, at a minimum, different actions of Kuhn occur across multiple hardware domains. For example, in the present invention, the functionality/interface actions occur within the mobile terminal itself. In Kuhn, functionality is spread among a device, a computer and a network.

Further, Kuhn does not disclose the interface component in the form of software instructions adapted to be loaded and stored in a computer readable medium (of the mobile terminal of the present invention). The Examiner cites col. 5, line 65 to col. 6, line 6:

Rather, the driver device 202 is generic to various carriers and or service providers in a given wireless network. For example, the driver device 202 may be generally operable in all networks in France, but the internal components are not programmed to handle communications with one specific carrier and or service provider. In one embodiment, the driver device 202 is built into the memory device 120 or, more specifically, into a read-only memory (ROM) portion of the memory device 120.

However, the foregoing must be understood in the context of the Kuhn invention wherein different actions of Kuhn occur across multiple hardware domains or nodes, whereas in the present invention, the functionality occurs within the mobile terminal itself. In Kuhn, functionality is spread among a device, a computer and a network. Thus, it is clear that there is more than a single interface that is used to implement the Kuhn invention.

Further, Kuhn does not disclose the interface component executed by a processor; and plug-in software in the form of software instructions adapted to be loaded and stored in a computer readable medium and executed by the processor of the mobile terminal for use by the application domain software for modifying the functionality of the software services component of the mobile terminal platform domain via the at least one interface. The Examiner cites the plug-in 204 of Fig. 2; col. 6, lines 31-47 and lines 57-65 and col. 7, lines 29- 61:

To deactivate dc-register a computer 100, the driver device 202 searches through all plug-in devices 204 in the computer 100. If no plug-in devices 204 are found, deactivation is complete because without a plug-in device, a computer 100 is not network activated. In searching for plug-in devices 204, the computer 100 determines which network plug-in device is responsible for deactivation. The driver device 202 launches the found plug-in with a deactivation code. Note that the deactivation code must be accompanied by appropriate parameters to denote the desired plug-in device for deactivation. If the plug-in device also contains the service provider's registration (or de-registration) set of codes, the driver device 202 must determine whether to launch the plug-in device 204 with a "dc-register" code first. In one embodiment, the deactivation code is called by a registration file 208 that is in the process of deactivating a component of service.

The plug-in device 204 may be configured to assist in establishing activation and registration, by executing functions such as the following: setting system features for the network activation upon resetting of the GARF 200; network activation deactivation with a particular carrier and registration dc-registration with a particular service provider; maintenance of account identification for a particular carrier and a particular service provider; and initiating appropriate user interfaces during error conditions.

As a secondary function, the plug-in device 204 provides error and warning code user interfaces, which are tailored to a particular verbal language (e.g., French, English, German, Spanish). In another embodiment, the plug-in device 204 is designed to be compatible with multiple carriers and service providers in a given network. For example, the plug-in device 204 may be multiplexed to receive a call from the driver device 202, and the plug-in device 204 may then switch between different carriers, depending on the call. In still another embodiment, multiple, different plug-in devices 204 may be installed in the computer 100, for example, to allow the user to have a choice of various carriers and/or service providers. In yet another embodiment, a general plug-in device 204 is provided, for example, a plug-in device compatible with an English language wireless network covering all of North America. Alternatively, a more specific plug-in device 204 may be installed to handle carriers and service providers, for example, in just the western United States.

Further, the different plug-in devices 204 may be configured to be compatible with each other. Regardless of the particular configuration of the plug-in devices 204, however, a plug-in device 204 is a component that can be readily redesigned after a computer 100 has been

manufactured. Such flexibility of a plug-in device 204 is useful because hardware and software components in a computer 100 may be entrenched within the computer's operating system (e.g., the ROM). Thus, the plug-in device 204 provides a component for the computer 100 that facilitates changing the computer's compatibility for different carriers and/or service providers.

Again, the foregoing must be understood in the context of the Kuhn invention wherein different actions of Kuhn occur across multiple hardware domains or nodes (requiring the use of several protocol stacks for each node or hardware domain), whereas in the present invention, the functionality occurs within the mobile terminal itself. In Kuhn, functionality is spread among a device, a computer and a network.

The Examiner only tangentially references Allor. Allor discloses a system and method for displaying plug-in supplied content in a browser-embedded window. As noted above, the functionalities of Kuhn occur across 3 domains--a PDA, a computer and a network, whereas all of the functionality of the present invention occurs in a mobile terminal. Clearly, the elements of Allor and Kuhn in combination do not disclose or suggest the present invention.

CONCLUSION

In view of the foregoing remarks, the Applicants believe all of the claims currently pending in the Application to be in a condition for allowance. The Applicants, therefore, respectfully request that the Examiner withdraw all rejections and issue a Notice of Allowance for claims 23-44.

The Applicants request a telephonic interview if the Examiner has any questions or requires any additional information that would further or expedite the prosecution of the Application.

Respectfully submitted.



Michael Cameron
Registration No. 50.298

Date: March 11, 2008

Ericsson Inc.
6300 Legacy Drive, M/S EVR 1-C-11
Plano, Texas 75024

(972) 583-4145
michael.cameron@ericsson.com